AMENDMENTS TO THE CLAIMS

1. (Currently Amended) A simultaneous multithreaded processor system comprising:

a first multiplexer associated ~~to~~ with ~~a first thread of~~ instruction pointers of a first thread; and

a second multiplexer associated ~~to~~ with ~~a second thread of~~ instruction pointers of a second thread;

said first and second multiplexers to provide said instruction pointers of said first and second threads for execution in said processor;

first and second storage elements coupled to said respective first and second multiplexers, wherein:

one of the first and second threads is active while the other of said first and second threads is inactive; and

said instruction pointers for the active thread are delivered to processor logic and ~~the~~ said instruction pointers for the inactive thread are delivered to ~~a~~ said storage element for delivery to the processor logic when the inactive thread becomes the active thread.

2. (Original) The system of claim 1, further comprising a common multiplexer coupled between said first and second multiplexer and processor logic.

3. (Original) The system of claim 2, wherein the common multiplexer receives instruction pointer data sequentially from the first multiplexer and the second multiplexer by utilizing a time-multiplexing protocol.

4. (Original) The system of claim 3, wherein the time-multiplexing protocol is a 'round-robin' protocol.

5. (Original) The system of claim 1, wherein the first multiplexer and the second multiplexer are priority multiplexers.

6. (Original) The system of claim 5, wherein the first multiplexer and the second multiplexer receive instruction pointer information and data from a plurality of stages in a processor pipeline.

7. (Original) The system of claim 6, wherein the first multiplexer and the second multiplexer receive instruction pointer information and data from re-steer logic at the plurality of stages in the processor pipeline.

8. (Original) The system of claim 7, wherein the first multiplexer and the second multiplexer pass the instruction pointer information and data to the common multiplexer with a pre-determined priority.

9. (Original) The system of claim 1, wherein the storage element is a flip-flop device.

10. (Currently Amended) A method for a simultaneous multithreaded processor system, comprising the steps of:

associating a first multiplexer ~~to~~ with instruction pointers of a first thread ~~of instruction pointers~~;

associating a second multiplexer ~~to~~ with instruction pointers of a second thread ~~of instruction pointers~~;

providing, by said first and second multiplexers, said instruction pointers of said first and second thread for execution in said processor;

coupling first and second storage elements to said respective first and second multiplexers;

establishing one of the first and second threads as active and the other of said first and second threads as inactive;

delivering ~~the~~ said instruction pointers for the active thread to processor logic; and

delivering ~~the~~ said instruction pointers for the inactive thread to ~~a~~ said storage element for delivery to the processor logic when the inactive thread becomes the active thread.

11. (Original) The method of claim 10, further comprising:

coupling a common multiplexer between said first and second multiplexer and processor logic.

12. (Original) The method of claim 11, wherein the common multiplexer receives instruction pointer data sequentially from the first multiplexer and the second multiplexer by utilizing a time-multiplexing protocol.

13. (Original) The method of claim 12, wherein the time-multiplexing protocol is a 'round-robin' protocol.

14. (Original) The method of claim 10, wherein the first multiplexer and the second multiplexer are priority multiplexers.

15. (Original) The method of claim 14, wherein the first multiplexer and the second multiplexer receive instruction pointer information and data from a plurality of stages in a processor pipeline.

16. (Original) The method of claim 15, wherein the first multiplexer and the second multiplexer receive instruction pointer information and data from re-steer logic at the plurality of stages in the processor pipeline.

17. (Original) The method of claim 16, wherein the first multiplexer and the second multiplexer pass the instruction pointer information and data to the common multiplexer with a pre-determined priority.

18. (Original) The method of claim 10, wherein the storage element is a flip-flop device.

19. (Currently Amended) A simultaneous multithreaded processor system comprising:

a first multiplexer associated ~~to~~ with instruction pointers of a first thread ~~of instruction pointers~~; and a second multiplexer associated ~~to~~ with instruction pointers of a second thread ~~of instruction pointers~~;

said first and second multiplexers to provide said instruction pointers of said first and second threads for execution in said processor;

first and second storage elements coupled to said respective first and second multiplexers, wherein:

one of the first and second threads is active while the other of said first and second threads is inactive;

said instruction pointers for the active thread are delivered to processor logic and ~~the~~ said instruction pointers for the inactive thread are delivered to ~~a~~ said storage element for delivery to the processor logic when the inactive thread becomes the active thread; and

a common multiplexer coupled between said first and second multiplexer and processor logic that receives instruction pointer data sequentially from the first multiplexer and the second multiplexer by utilizing a time-multiplexing protocol.

20. (Original) The system of claim 19, wherein the first multiplexer and the second multiplexer receive instruction pointer information and data from a plurality of stages in a processor pipeline.

21. (Original) The system of claim 20, wherein the first multiplexer and the second multiplexer receive instruction pointer information and data from re-steer logic at the plurality of stages in the processor pipeline.

22. (Original) The system of claim 19, wherein the first multiplexer and the second multiplexer pass the instruction pointer information and data to the common multiplexer with a pre-determined priority.